رگرسیون خطی

linear model

برای انجام رگرسیون خطی از تابع lm( ) استفاده می‌کنیم:

lm(formula , data, ...)

formula : مدل رگرسیونی است که می‌خواهیم به داده‌ها برازش دهیم. به عنوان مثال به صورت y ~ X1+X2 می‌نویسیم که در آن X1 و X2 متغیرهای توضیحی و y متغیر پاسخ است.

data : معرف مجموعه داده‌هایی است که متغیرها در آن قرار دارند.

مثال: مجموعه داده‌های Highway1 موجود در R را در نظر بگیرید:

```
> rm(list = ls())
> Highway1
> head(Highway1)
```

head(Highway1) ← چند سطر اول مجموعه داده را نشان می‌دهد

rate : نرخ تصادفات 1973 در ایالت مینسوتای آمریکا در هر میلیون مایل جاده

len : طول بزرگراه به مایل

adt : متوسط تعداد ترافیک روزانه به 1000

trks : نسبت کامیون‌ها به تعداد کل خودروها

sigs1 : تعداد سیگنال‌ها در هر مایل جاده

slim : محدودیت سرعت

shld : پهنای شانه جاده

lane : تعداد کل لاین‌های جاده

acpt : تعداد نقاط دسترسی در هر مایل

itg : تعداد تقاطعات نوع آزادراه در هر مایل

lwid : عرض لاین‌ها

htype : نوع جاده

```
> mymodel = lm( rate ~ acpt + slim + len + shld , Highway1)
```

```
> mymodel  # or print(mymodel)
```

مدل برازش داده شده یعنی برآوردهای ضرایب رگرسیونی را نمایش می‌دهد.

$$\hat{Y}_i = 9.17574 + 0.10156\, acpt_i + 0.09808\, slm_i - 0.07817\, len_i - 0.01099\, shld_i$$

```
> summary(mymodel)
```

خلاصه‌ای از نتایج برازش مدل شامل آماره‌های توصیفی از باقیمانده‌های مدل، برآوردهای ضرایب رگرسیونی به همراه برآورد انحراف استاندارد و آماره‌های t و p-مقدارهای آزمون‌های فرضیه‌ی ضرایب رگرسیونی جزئی. همچنین $\sqrt{MSE}$، R^2 و R^2_{adj} و آماره F جدول ANOVA:

$$\underset{\sim}{Y} = X\underset{\sim}{\beta} + \underset{\sim}{\epsilon} \quad \leftarrow \quad Y_i = \beta_0 + \beta_1 x_{i1} + \dots + \beta_k x_{ik} + \epsilon_i \quad i=1,\dots,n$$

$$\underset{\sim}{Y} = \begin{bmatrix} Y_1 \\ \vdots \\ Y_n \end{bmatrix} \quad X = \begin{bmatrix} \underset{\sim}{1} & \underset{\sim}{x}_1 & \cdots & \underset{\sim}{x}_k \end{bmatrix} \quad \underset{\sim}{\beta} = \begin{bmatrix} \beta_0 \\ \vdots \\ \beta_k \end{bmatrix}$$

$$F = \frac{SSR/k}{MSE}$$

$F > F_{1-\alpha}(k, n-p) \Rightarrow$ رد H_0

$$\begin{cases} H_0: \beta_1 = \dots = \beta_k = 0 \\ H_1: o.w. \end{cases}$$

آزمون معناداری کل مدل

$$\hat{\underset{\sim}{\beta}} = (X'X)^{-1}X'\underset{\sim}{Y} \qquad Var(\hat{\underset{\sim}{\beta}}) = \sigma^2 (X'X)^{-1} = \sigma^2 C$$

$$Var(\hat\beta_j) = \sigma^2 C_{jj} \quad , \quad \widehat{Var(\hat\beta_j)} = MSE\, C_{jj}$$

$$Cov(\hat\beta_j, \hat\beta_r) = \sigma^2 C_{jr} \quad , \quad \widehat{Cov(\hat\beta_j, \hat\beta_r)} = MSE\, C_{jr}$$

$$R^2 = \frac{SSR}{SSY} = 1 - \frac{SSE}{SSY} \qquad MSE = \frac{SSE}{n-p} \qquad R^2_{adj} = 1 - \frac{MSE}{MSY} = 1 - \frac{SSE/n-p}{SSY/n-1}$$

$$\hat\beta_0 \quad \sqrt{MSE\, C_{00}} \quad T = \frac{\hat\beta_0 - 0}{\sqrt{MSE\, C_{00}}} \quad p\text{-value} = P_{H_0}(|T| > |T_0|)$$

$$\hat\beta_1 \quad \sqrt{MSE\, C_{11}} \quad T = \frac{\hat\beta_1 - 0}{\sqrt{MSE\, C_{11}}} \quad p\text{-value} = P_{H_0}(|T| > |T_0|)$$

$$\vdots$$

$$\begin{cases} H_0: \beta_1 = 0 \\ H_1: \beta_1 \neq 0 \end{cases}$$

آزمون‌های فرضیه‌ی ضرایب رگرسیون جزئی

* اگر H_0 رد نشود یعنی به متغیر x_1 در حضور سایر متغیرهای توضیحی نیازی نیست.

$$\underset{\sim}{t} \quad T^2 = F = \frac{\hat\beta_1^2}{MSE\, C_{11}} \qquad p\text{-value} = P_{H_0}(F > F_0)$$

```
> anova(mymodel)
```

مشابه خروجی قبل است با این تفاوت که آماره‌های F را به جای T را. مقادیر R^2، R^2 تعدیل شده را گزارش نمی‌دهد.

برای به دست آوردن SSR، SSE و SSY به طریق زیر عمل می‌کنیم:

```
> SSE = anova(mymodel)["Residuals", "Sum Sq"]

> n = length(Highway1$rate)

> SSY = t(Highway1$rate) %*% Highway1$rate - n *
                                        (mean(Highway1$rate))^2

> SSR = SSY - SSE
```

برای محاسبه $\hat{Y}_i$ ها، $\hat{\varepsilon}_i$ ها و $\hat{\beta}_j$ ها به ترتیب داریم:

```
> mymodel$fitted.values # or fitted.values(mymodel) or
                                                  fitted(mymodel)

> mymodel$residuals # or residuals(mymodel) or
                                                  resid(mymodel)

> mymodel$coefficients # or coefficients(mymodel)
```

برای به دست آوردن برآورد فاصله‌ای ضرایب رگرسیونی:

```
> confint(mymodel, level = 0.9)
```

$$\hat{\beta}_j \pm t_{1-\alpha/2}(n-p)\sqrt{MSE\, C_{jj}}$$

برای به دست آوردن برآورد ماتریس $Var(\hat{\beta})$ داریم:

```
> vcov(mymodel)
```

→ ماتریسی شامل واریانس‌ها و کوواریانس‌های $\hat{\beta}_j$ ها.

رسم نمودار خط رگرسیون برازش داده شده:

درصورتی که مدل شامل یک متغیر توضیحی یا دو متغیر توضیحی باشد می‌توان به ترتیب خط و صفحه برازش داده شده را رسم کرد.

مثال:

```
> mymodel1 = lm(rate ~ acpt, Highway1)
> plot(Highway1$acpt, Highway$rate)
> abline(mymodel1, col = "red", lwd = 2)
```

مثال: ابتدا پکیج های car و rgl را نصب کنید سپس

```
> library(car)
> scatter3d(Highway1$rate ~ Highway1$acpt + Highway1$slim ,
            revolutions = 3, speed = 0.5 ,
            grid = F)
```

اگر ننویسیم تصویر ثابت می‌ماند.

چنانچه بخواهیم از تمام متغیرهای موجود که در مجموعه داده، به عنوان متغیر توضیحی مدل استفاده کنیم:

```
> mymodel2 = lm(rate ~ . , Highway1)
> mymodel2
```

برای متغیرهای رشته‌ای، متغیرهای نشانگر در نظر می‌گیرد.

اگر بخواهیم رگرسیون عبوری از مرکز برازش دهیم:

```
> mymodel3 = lm(rate ~ acpt + slim + len + shld - 1 ,
                                        Highway1)
> mymodel3
```

می‌خواهیم فواصل اطمینان برای میانگین های پاسخ و فواصل پیش‌بینی

برای این نقطه داریم:

```
> predict(mymodel, newdata = list(acpt = 4.5, slim = 50, lens = 6
       , shld = 11 ), interval = "confidence", level = 0.95)
```

← برای فواصل اطمینان میانگین پاسخ

$$\hat{Y}_0 \pm t_{1-\alpha/2}(n-p)\sqrt{MSE\left(\frac{1}{n} + x_0'(X'X)^{-1}x_0\right)} \qquad x_0 = \begin{bmatrix} 1 \\ x_{01} \\ \vdots \\ x_{0k} \end{bmatrix}$$

```
> predict(      ,          ,          ,
          ,      , interval = "prediction" , level = 0.95)
```

← برای فواصل پیش‌بینی

$$\hat{Y}_0 \pm t_{1-\alpha/2}(n-p)\sqrt{MSE\left(1 + \frac{1}{n} + x_0'(X'X)^{-1}x_0\right)}$$

چنانچه مدل رگرسیون خطی ساده باشد، می‌توان نمودار این فواصل را نیز رسم کرد:

```
> n = nrow(Highway1)

> mint = predict(lm(rate ~ acpt, Highway1), interval="confidence",
                                                   level=0.95)

> myfit0 = mint[1:n]

> mylwr0 = mint[(n+1):(2*n)]

> myupr0 = mint[(2*n+1):(3*n)]
```

* ساختار mint اینگونه ایجاد کرده‌ایم atomic است و بعبارتی ساختار list ندارد و بنابراین

$ را نمی‌توان بکار برد

```
> pint = predict( , , interval="prediction",
                                         level=0.95)
> myfit1 = pint[1:n]

> mylwr1 = pint[(n+1):(2*n)]

> myupr1 = pint[(2*n+1):(3*n)]
```

```
> X = Highway1$acpt ; Y = Highway1$rate
> plot(X, Y) ~~pch=20~~        →        نمودار پراکنش
```

رسم فواصل اطمینان

```
> lines(sort(X), myfit0[order(X)], col='black', lwd=2)

> lines(sort(X), mylwr0[order(X)], col='blue', lwd=2,
                                                  lty=2)

> lines(sort(X), myupr0[order(X)], col='blue', lwd=2,
                                                  lty=2)
```

```
>                 myfit1
>                 mylwr1
>                 myupr1
```

تکرار خطوط بالا اینبار با

جهت رسم فواصل پیش‌بینی